Data Tracking for ReMeltRadar

Version 0.1 - 2022-01-13

Data within `/Raw`, `/Untouched`, `/QGis` and `/Log` are untracked by Git to avoid uploads of large data files and immediately into public domain.

To ensure that data can be shared between all contributors, each folder contains a `catalogue.csv` file which lists files contained within each folder.

If you add a new file to any of these folders then update `catalogue.csv`, which can be used by other contributors to produce a report of any data they may be missing.

File Format for `catalogue.csv`

The `catalogue.csv` file should be UTF-8 encoded and contain the following comma separated headings in the first line

```
filename,size,owner,date_added,comment
```

There should be one entry per file with the following values. Each row should contain at least **filename**, other fields are optional.

filename: name of the file to be catalogued.

size: file size in bytes

owner: name of the person to collect the file from.

date_added: YYYY-MM-DD formatted date of the day the file was added to `catalogue.csv`.

comment: brief description displayed in the missing files report or for reference purposes.

An example dataset is as follows:

```
filename,size,owner,date_added,comment
an_apres_survey.dat,32052220,Reza,2022-01-13,some data from an ApRES survey
another_file.txt,1603,,,
pulseEKKO_project.gpz,17442090,Inka,2022-01-09,
...
```

Generate Missing File Report

To generate a missing file report, the following command should be called from the project root using the option `-r`:

```
python bin/data.py -r /path/from/root/to/data_folder
```

which will generate a log file named `data_report_YYYYMMDD_HHMMSS.log` in `/Log` for `data_folder`.

If you want do this for all subfolders, then you can use the option `-s`

```
python bin/data.py -r -s /path/from/root/to/data_folder
```

Automatically Update `catalogue.csv`

To automatically update the `catalogue.csv` file in a given folder, the following command should be called from the project root using the option `-u`:

```
python bin/data.py -u /path/from/root/to/data_folder
```

which will update the file `/path/from/root/to/data_folder/catalogue.csv` to include any files which are not currently included. If you want to specify the owner, then passing an additional argument will set the `owner` for each record.

```
python bin/data.py -u /path/from/root/to/data_folder owner_name
```

Similarly, to update catalogue files in add subfolders, use the `-s` flag.

```
python bin/data.py -u -s /path/from/root/to/data_folder [owner_name]
```